REMARKS

Claims 1-20 are presented for further examination. Claims 1, 8, 9, 11, and 13-20 have been amended.

In the Office Action mailed November 3, 2004, the Examiner objected to the oath/declaration as defective because it did not state that the person making the oath or declaration acknowledges the duty to disclose to the Office all information all information known to the person to be material to patentability as required under 37 C.F.R. § 1.56. Applicant is preparing a new declaration which will be submitted in due course.

The Examiner raised objections to the claims because of informalities therein, as noted in paragraphs 4-31 and rejections as noted in paragraphs 32-44 of the Office Action. Applicant has amended the claims as suggested by the Examiner to overcome the objections and rejections. No new matter has been entered. In addition, corresponding revisions have been made to the specification and abstract, as set forth above. Again, no new matter has been added. Approval and entry of these amendments is respectfully requested.

Claims 1-20 were rejected under 35 U.S.C. § 102(e) as anticipated by U.S. Patent No. 6,219,830 ("Eidt et al.").

Applicant respectfully disagrees with the basis for the rejection and requests reconsideration and further examination of the claims.

The disclosed embodiment of the invention will now be discussed in comparison to the applied reference. Of course, the discussion of the disclosed embodiment and the discussion of the differences between the disclosed embodiment and the subject matter described in the applied reference do not define the scope or interpretation of any of the claims. Rather, such discussed differences merely assist the Examiner in appreciating important claim distinctions discussed thereafter.

The present invention provides a method for linking a plurality of object code modules to form an executable program, and a resulting computer product is also provided. The present invention utilizes novel relocation instructions during the linking process that avoids the same calculation being passed to a linker many times over. More particularly, in the present

method and resulting computer product, symbol attributes are used to rewrite the code during the linking process.

Eidt et al. describe at column 1, lines 24-36, the steps in passing from a computer program written in source code to execution of the program on a computer system to be as follows:

(1) one or more source code modules are passed through a compiler or assembler that generates one or more object code files as an output;

(2) a linker routine, which is either a separate program or is part of the compiler, combines the source code modules into a single output file, known as an "executable" object code file; and

(3) one or more executables are then loaded together into memory by a loader program, and control is then transferred to a start address to initiate program execution.

As discussed above, the present invention relates to the linker routine in step 2, while the relocatable object code format and method for loading same into the computer system provided by Eidt et al. relates to the linker program of step 3. This is clear from the summary of the invention section of Eidt et al. at column 4, lines 54-67, which specifies that the invention takes advantage of certain characteristics of executable object code files to drastically reduce the number of bytes of relocation information that are required per relocation. In order to obtain the desired result, Eidt et al. disclose a loader program as described at column 9, line 34 through column 19, line 33, which may be applied to a re-locatable object code file format produced by a compiler prior to linking (step 1) at column 19, lines 26-33.

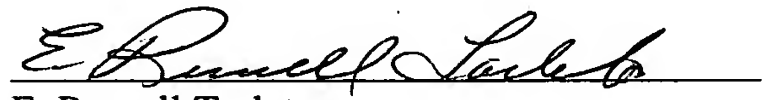
Independent claims 1, 11, and 13-20 have all been amended to clearly recite a method for a computer program product for linking a plurality of the object code modules to form an executable program. Nowhere in Eidt et al. is there any disclosure or suggestion that relates to forming an executable program using a linker, and in particular to use of symbol attributes when forming an executable program. Rather, Eidt et al. only disclose performing relocations before or after the linking process. As such, the present invention is clearly novel and nonobvious over the teachings of Eidt et al.

Applicant further notes that the Eidt et al. reference was cited in the European standard search report as being in category A, *i.e.*, of only technological background to the presently claimed invention. Thus, it is clear that Eidt et al. do not relate to the optimization of the linking process and in fact provide no detailed description of the linking process and how it might be achieved as set forth in claims 1-20.

In view of the foregoing, applicant respectfully submits that claims 1-20 are clearly in condition for allowance. In the event the Examiner finds minor informalities that can be resolved by telephone conference, the Examiner is urged to contact applicant's undersigned representative by telephone at (206) 622-4900 in order to expeditiously resolve prosecution of this application. Consequently, early and favorable action allowing these claims and passing this case to issuance is respectfully solicited.

Respectfully submitted,

SEED Intellectual Property Law Group PLLC



E. Russell Tarleton
Registration No. 31,800

ERT:jl

Enclosure:
Postcard

701 Fifth Avenue, Suite 6300
Seattle, Washington 98104-7092
Phone: (206) 622-4900
Fax: (206) 682-6031

558521_1.DOC